

**RESPONSE UNDER 37 C.F.R. § 1.116
EXPEDITED PROCEDURE
GROUP ART UNIT 2192**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 10/679,870	§	Examiner:	Yigdall, Michael J.
Filed: October 6, 2003	§	Group/Art Unit:	2192
Inventor(s):	§	Atty. Dkt. No:	5150-77600
Patrick Williams, David Rohacek	§		
	§		
	§		
	§		
	§		
Title: PARAMETER	§		
ASSISTANT FOR	§		
AUTOMATICALLY	§		
SPECIFYING	§		
PARAMETER VALUES	§		
FOR A SOFTWARE	§		
PROGRAM	§		

REQUEST FOR PRE-APPEAL BRIEF REVIEW

ATTN: BOX AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request. This request is being filed with a Notice of Appeal. The review is requested for the reason(s) stated below. Please note that for brevity, only the primary arguments directed to the independent claims are presented, and that additional arguments, e.g., directed to the subject matter of the dependent claims, will be presented if and when the case proceeds to Appeal.

Claims 1, 9, 11, 13, 20-24, 26, 27, 31 and 32 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,026,233 to Shulman et al. (hereinafter "Shulman") in view of U.S Patent No. 5,784,275 to Sojoodi et al. (hereinafter "Sojoodi"). Applicant respectfully traverses this rejection.

Claim 1 recites in pertinent part:

programmatically determine one or more valid parameter values for the first parameter of the first function call by invoking software for a measurement device in order to determine one or more hardware resources of the measurement device, wherein each of the one or more valid parameter values represents a respective hardware resource of the one or more hardware resources; (emphasis added)

Applicant respectfully submits that Shulman and Sojoodi, taken either singly or in combination, do not teach these limitations in combination with the other limitations recited in claim 1.

With respect to the limitation of invoking software for a measurement device in order to determine one or more resources of the measurement device, the Examiner cites Sojoodi at Col. 7, lines 3-17, where Sojoodi teaches querying an object manager for a list of VISA (Virtual Instrument Software Architecture) classes associated with an instrument. The VISA classes of the instrument correspond to the possible VISA interface types of the instrument. On p. 2 of the Office Action the Examiner states that:

“querying the object manager for a list of classes of the instrument, where the classes correspond to possible VISA interface types of the instrument” (see, for example, column 7, lines 3-17) represents querying the object manager for a list of one or more hardware resources of the instrument.

Applicant respectfully disagrees and submits that the VISA classes are not hardware resources of the instrument. The web document located at the URL <http://www.ni.com/support/visa/vintro.pdf> entitled, “LabVIEW VISA Tutorial” provides additional information describing the VISA programming interface and VISA classes. (Applicant notes that Sojoodi’s system is described in the context of the LabVIEW graphical programming environment. See for example, Col. 12, lines 4-16.) As described in the section titled “VISA Overview” on p. 1:

VISA is a standard I/O language for instrumentation programming. VISA by itself does not provide instrumentation programming capability. VISA is a high-level API that calls into lower level drivers. ... VISA is capable of controlling VXI, GPIB, or Serial instruments and makes the appropriate driver calls depending on the type of instrument being used.

And on p. 2:

One of VISA’s advantages is that it uses many of the same operations to communicate with instruments regardless of the interface type. For example, the VISA command to write an ASCII string to a message-based instrument is the same whether the instrument is Serial, GPIB, or VXI. Thus, VISA provides interface independence. This can make it easy to switch interfaces and also gives users who must program instruments for different interfaces a single language they can learn.

VISA classes are described on p. 6:

The VISA Class is a grouping that encapsulates some or all of the VISA operations. INSTR is the most general class that encompasses all VISA operations. In the future, other classes may be added to the VISA specification. Currently the VISA Class does not need to be included as part of the instrument descriptor. Nevertheless, this should be done to ensure future compatibility. Currently, most applications simply use the INSTR class.

Thus, VISA defines an object-oriented style interface for communicating with instruments in an interface-independent manner, and a VISA class is simply a grouping of particular VISA operations. Applicant thus respectfully submits that the Examiner's interpretation of a VISA class as a hardware resource of an instrument is erroneous.

Applicant notes that claim 1 further recites:

receive user input to the graphical user interface to select a first parameter value from the one or more valid parameter values, wherein the first parameter value represents a first hardware resource of the measurement device; and

automatically modify the first function call displayed in the source code of the software program by including the first parameter value in the first function call in response to the user input selecting the first parameter value, wherein automatically including the first parameter value in the first function call aids a user in modifying the first function call to reference the first hardware resource of the measurement device. *(emphasis added)*

As noted above, the Examiner has interpreted the recited "one or more parameter values" as parameters representing the VISA classes associated with an instrument. Thus, the recited "first parameter" represents a particular one of the VISA classes of the instrument. Thus, automatically modifying the first function call by including the first parameter value in the first function call as recited in the claim results in the insertion of a parameter which represents a particular VISA class. However, this does not aid a user in modifying the first function call to reference a first hardware resource of a measurement device, as recited in the claim. Instead, it would modify the first function call to reference a particular VISA class. Applicant respectfully submits that the combination of Shulman and Sojoodi does not teach the concept of aiding a user in modifying a function call to reference a first hardware resource of a measurement device, where the first hardware resource is determined by invoking software for the measurement device, as recited in claim 1.

Applicant thus respectfully submits that independent claim 1 is patentably distinct over the cited art for at least the reasons set forth above. Inasmuch as the other independent claims

recite limitations similar as those of claim 1 discussed above, Applicant respectfully submits that these claims are also patentably distinct over the cited art.

CONCLUSION

In light of the foregoing amendments and remarks, Applicant submits the application is now in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above-referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. The Commissioner is hereby authorized to charge any fees which may be required or credit any overpayment to Meyertons, Hood, Kivlin, Kowert & Goetzel P.C., Deposit Account No. 50-1505/5150-77600/JCH. Also filed herewith is the following item: Notice of Appeal

Respectfully submitted,

/Jeffrey C. Hood/

Jeffrey C. Hood, Reg. #35198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: 2009-05-26 JCH/JLB